PRELIMINARY

2920/2921
SIGNAL PROCESSOR

- **2920—User-Programmable/Erasable EPROM Program Memory (T_{CYC} = 600 ns)**
- **2921—Mask-Programmable ROM Program Memory (T_{CYC} = 400 ns)**

- **Real Time Digital Processing of Analog Signals**
- **Stability and Accuracy of Digital Processing**
- **Nominal Signal Bandwidths from D.C. to 10 KHz**
- **Multiple Analog Inputs and Outputs**
- **On Chip A/D and D/A**
- **On Chip Program Memory**
- **On Chip Scratch Pad Memory**
- **Analog and/or TTL Output Waveforms, User Selectable**
- **Intellec® Development System Support:**
  - **— Compiler**
  - **— Assembler**
  - **— Simulator**

The Intel 2920 Signal Processor is a programmable, single chip analog and digital signal processor specifically designed to replace analog subsystems in real time processing applications. The 2921 is an improved performance mask programmable (ROM) version of the 2920 which is pin compatible with the 2920. Its instruction set plus the high precision (25 bits) digital arithmetic logic unit provides the capability to implement very complex subsystems. Typical functions performed by the 2920 include: Lowpass and Bandpass filters with up to 20 complex pole and/or zero pairs; Threshold Detectors; Limiters; Rectifiers; up to 25-bit multiplication and division; approximations to nonlinear functions such as square law and logarithm; logical operations; input and Output multiplexing of signals; logical outputs for decision-type processing; and analog outputs for multifrequency oscillators, waveform generators, etc. In addition, several 2920's may be cascaded for very complex processing applications with no loss in throughput rate.

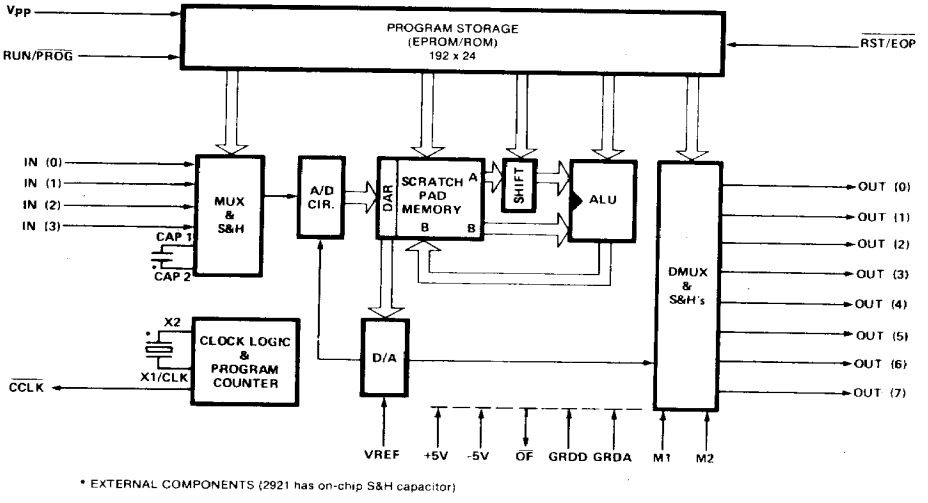


Figure 1. Functional Block Diagram (Run Mode)

Table 1. Pin Description (Run Mode)

Symbol	Function
OUT	Analog Output (0-7)
GRDA	Analog Ground
CAP_1 & CAP_2	External capacitor connections for the input signal sample and hold circuit. Not required for 2921.
VREF	Reference Voltage
IN	Analog Input (0-3)
V_{BB}	Most negative power pin set at −5 volts during run mode (different voltage in program mode).
X_2/CLK	Clock input when using external clock signals; oscillator input for external crystal when using internal clock.
X1	Oscillator input for external crystal when using internal clock.
GRDD	Digital ground.
V_{CC}	5 volts in run mode.
$\overline{CCLK}$	Internal fetch cycle clock output. The falling edge designates the START of a new PROM fetch cycle. $\overline{CCLK}$ is 1/16 of X_2/CLK rate.
RUN/$\overline{PROG}$	Mode control tied to GRDD in run mode.
$\overline{RST}$/$\overline{EOP}$	Reset Input and/or end of program output

Symbol	Function
$\overline{OF}$	Indicates an overflow in the current ALU operation (open drain, active low).
Vpp	Programming Power (0 volts for run mode)
M1, M2	Output mode control (See Table 8)

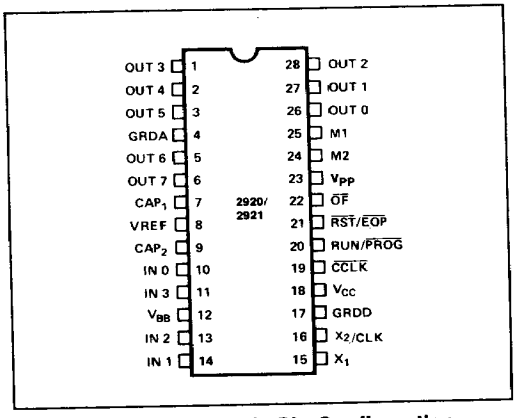


Figure 2. Run Mode Pin Configuration

Table 2. Pin Description (Program/Verify Mode)

Symbol	Function
D0,D1,D2,D3	4 pins carrying EPROM program data for both input and output (open drain, active low output; active high input).
V_{B1},V_{B2}, V_{B3},V_{B4}	Digital ground in PROGRAM mode (different voltage for RUN mode).
V_{S1},V_{S2},V_{S3}	+5 volts in PROGRAM mode (function changes for RUN mode).
$\overline{INCR}$	Input pulse increments the nibble (4-bits) counter in PROG mode (function changes in RUN mode).

Symbol	Function
Vpp	EPROM/ROM power pin, +5 volts for VERIFY mode. For 2920 program mode pin is +25 volts (different voltage in RUN mode).
PROG/$\overline{VER}$	Controls EPROM bi-directional data bus for verify (low) or program (high).
$\overline{RST}$	Input pulse resets nibble counter to position zero for start of programming.

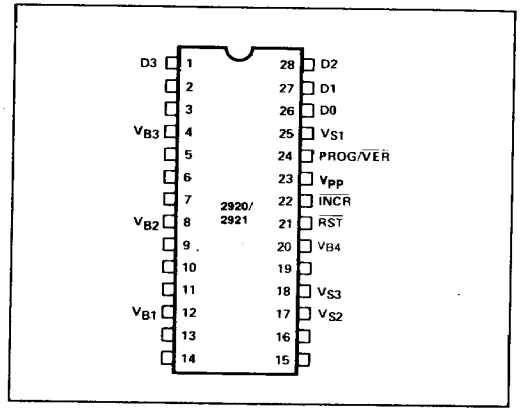


Figure 3. Program Mode Pin Configuration

FUNCTIONAL DESCRIPTION

The Intel 2920 is a programmable, single chip analog and digital signal processor which has been designed specifically to replace analog systems in real-time processing applications. The 2920 operates its analog circuitry simultaneously with the digital circuitry thereby achieving the efficiency and speed needed for real-time operation. Digital Circuitry includes: EPROM/ROM program storage, RAM scratch pad memory, clock and timing circuitry, binary scaler, and the arithmetic logic unit (ALU). The analog circuitry is composed of 4 analog inputs, an input multiplexer, an input sample and hold (S&H), A/D and D/A converters, an output multiplexer, 8 analog outputs, and buffered output S&H's.

Once the EPROM/ROM is programmed, the 2920 is ready for operation as an analog subsystem. The following signal flow and operations can be described with reference to the functional block diagram shown in Figure 1.

Clock and Timing Logic

The 2920/2921 can use an external clock or can generate its own clock with an external crystal placed across Pins 15 and 16. The program counter is incremented one instruction count for every 4 master clock cycles and continues to increment until it reaches a count of 191 or an EOP instruction is executed. Instructions are executed sequentially and no program jumps are provided. The sample rate is determined by the number of instructions in the program and the instruction cycle time. A 6.67MHz clock (600 nsec instruction cycle time) and a full 192 instructions will result in a sample rate of 8680Hz [clock rate (MHz) ÷ (4 x # of instructions)].

Program Storage and Control

The EPROM/ROM is made up of 192 words with 24-bits per word. Each 24-bit word contains 6 infields (see Table 3) which control the individual subsystems in the 2920.

RAM

The memory consists of a random-access read/write array organized as 40 words of 25-bits each. The address space is extended to provide constants and access to a register (DAR) for interfacing the memory-ALU with the analog conversion section. The RAM is a two port memory where the "A" location is Read Only and passes via a scaler to the ALU as one operand. The "B" location data passes to the ALU input as its second operand and the ALU result is written back to it. Both the RAM and the ALU represent data in two's complement format. All operations are performed in two's complement arithmetic. Program operations are simplified by assuming the binary point to the right of the sign bit.

An extended address space is used to generate constants within the program. It is accessed through the "A" port only and may be addressed using the last 16 locations of the "A" address field (i.e., "A" address 11XXXX). The constant is determined by the 4 least significant "A" address bits.

These 4 bits are treated as the 4 most significant bits at the input to the binary shifter. A sequence of extended addresses with shift operations can generate any constant up to 25 bits long.

The DAR is 9 bits wide and can be accessed in several ways. As a memory location, the DAR occupies the 9 most significant bit positions of the 25-bit word and can be accessed as "A" and/or "B" port. The DAR output is also tied directly to the D/A converter inputs and is used as a successive approximation register for A/D conversion under control of the analog function instruction fields. Each bit position of the DAR can also be selected and tested for conditional arithmetic operations.

Binary Shifter

The 2920 has a binary shifter between the memory port output and the ALU "A" operand input. This feature allows the "A" operand to be scaled by any magnitude between 2^{0} and 2^{-13} (left shift 2 to right shift 13). When a number is shifted right vacated bit positions are filled with the sign bit. (2's complement arithmetic shift). Shift op codes are shown in Table 4.

ALU

The Arithmetic-Logic Unit calculates a 25-bit result based on an operation performed on the scaled "A" and the "B" operands delivered from memory. The 25-bit result is written back into the "B" memory location near the end of the instruction cycle. The ALU has logic to accommodate the left shift scaling. For arithmetic operations, this logic is used to calculate a 25-bit result for normal operations and to maintain the sign bit when an overflow occurs. An overflow occurs only when the magnitude of the result is larger than the

largest number that can be stored in memory (25 bits). In that event, the result is set to the largest magnitude value with the correct sign. This overflow algorithm protects the continuity of the digitized analog signals and helps maintain the stability of the signal processing functions implemented. It is analogous to an overdriven amplifier going into saturation.

Instruction Set

The 2920 assembler uses the following program format to specify the 24-bit instruction word stored in the EPROM:

All processing subsystems are implemented using a combination of analog and digital instructions to input and output signals and/or data, and to realize the processing functions, respectively.

The analog input and output instructions are IN(K) and OUT(K), respectively. A sequence of IN(K) instructions followed by the sign conversion and amplitude conversion instructions CVTS and CVT(K) respectively are used to perform the input A/D conversion. A simple sequence of OUT(K) instructions is all that is needed to output a 9-bit amplitude on channel K.

Other analog instructions are the EOP instruction which resets the program counter, NOP which is simply a no-operation, and CNDS or CND(K) which are conditional operators which select and test a bit in the DAR for the conditional ADD or LDA instructions or define the destination of the carry bit for the conditional SUB instruction.

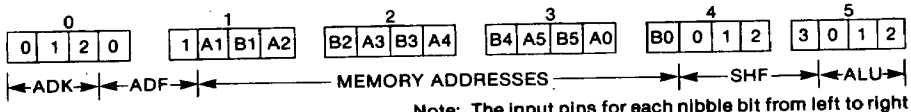
If used, the EOP instruction must be placed at a program location up to location 188 which is a multiple of four. The three instructions immediately following the EOP instruction will be executed.

The ALU arithmetic instructions are ADD,SUB,LDA which perform the operations of addition, subtraction, and data transfer respectively. When these instructions are conditioned, they may be used to perform multiplication or division by a variable or data dependent switching.

Other digital instructions include the absolute value ABS, the absolute value and add ABA, and the ideal limit instruction LIM.

These instructions and their corresponding op codes are detailed in Table 5.

Table 3. Nibble Organization for Loading Program



Note: The input pins for each nibble bit from left to right are D0, D1, D2, D3.

Table 4. Shift OP Codes

Operation	Mnemonic	Op Code 3	2	1	0	Scale Factor
Shift Right 13 Bits	R13	1	1	0	0	2^{-13}
Shift Right 12 Bits	R12	1	0	1	1	2^{-12}
⋮	⋮	⋮	⋮	⋮	⋮	⋮
Shift Right 1 Bit	R01	0	0	0	0	2^{-1}
No Shift	R00	1	1	1	1	1
Shift Left 1 Bit	L01	1	1	1	0	2
Shift Left 2 Bits	L02	1	1	0	1	4

Table 5. Instruction Set and OP Codes

Mnemonics		Op-Codes[1]			Operations			Notes
Code	Condition	ALU	ADF	ADK				
Digital Instructions		0,1,2	0,1	2,1,0				
ADD		011	[3]	[3]	$(Ax2^N) + B \longrightarrow B$			
SUB		101			$B - (Ax2^N) \longrightarrow B$			
LDA		111			$(Ax2^N) + 0 \longrightarrow B$			
XOR		000			$(Ax2^N) \oplus B \longrightarrow B$			
AND		100			$(Ax2^N) \cdot B \longrightarrow B$			
ABS		110			$\lvert(Ax2^N)\rvert \longrightarrow B$			
ABA[11]		001			$\lvert(Ax2^N)\rvert + B \longrightarrow B$			
LIM		010			$Sign(A) \rightarrow \pm$ F.S. $\rightarrow B$[4]			
ADD	CND( )[2]	011			$(Ax2^N) + B \longrightarrow B$	IFF DAR(K) = 1		
					$B \longrightarrow B$	IFF DAR(K) = 0		
SUB	CND( )[2][8]	101			$B - (Ax2^N) \longrightarrow B$	& CY $\rightarrow$ DAR(K)	IFF CY_P = 1	[5]
					$B + (Ax2^N) \longrightarrow B$	& CY $\rightarrow$ DAR(K)	IFF CY_P = 0	
LDA	CND( )[2]	111			$(Ax2^N) \longrightarrow B$		IFF DAR(K) = 1	
					$B \longrightarrow B$		IFF DAR(K) = 0	
ABA[11]	CND( )[9]	001			$(Ax2^N) + B \longrightarrow B$			
XOR	CND( )[9]	000			$(Ax2^N) \oplus B \longrightarrow B$			
Analog Instructions								
IN(K)		[7]	00	0-3	Signal Sample from Input Channel K			
OUT(K)			10	0-7	D/A to Output Channel K			
CVTS			00	6	Determine Sign Bit			
CVT(K)			01	0-7	Perform A/D on Bit K			
EOP			00	5	Program Counter to Zero			[6]
NOP			00	4	No Operation			
CND(K)			11	0-7	Select Bit K for Conditional Instructions			
CNDS			00	7	Select Sign Bit for Conditional Instructions			

Notes:
1. Op codes ALU and ADF are in binary notation, ADK is in decimal notation and represents the value "K" when appropriate.
2. CND( ) can be either CND(K) or CNDS testing amplitude bits or the sign bit of the DAR respectively.
3. Determined by analog instructions below.
4. B is set to full scale (F.S.) amplitude with the same sign as the "A" port operand.
5. The previous carry bit (CY_P) is tested to determine the operation. The present carry bit (CY) is loaded into the Kth bit location of the DAR. "Present carry (CY) is generated independent of overflow. It will represent the carry (CY) of a calculated 28-bit result."
6. EOP will also enable overflow detection if it was disabled during a program pass.
7. Determined by digital instructions above.
8. For SUB CNDS operation $\overline{CY} \rightarrow$ DAR(S).
9. Does not affect DAR. In this case, CND is used with XOR/ABA to enable/disable the ALU overflow saturation algorithm. An EOP instruction will also enable the ALU overflow saturation algorithm.
10. Clarification of CY_{OUT} sense for certain operations. For LDA, XOR, AND, ABS; $CY_{OUT} \rightarrow 0$.
11. Recommend that the ABS & ADD instructions be used in place of ABA. The ABA instruction typically runs 2 MHz slower. The saturation logic, however can be set and reset at full speed.

ANALOG INPUTS

An analog input is digitized using a successive approximate technique. The successive approximations are software controlled. This allows the flexibility to convert a given input to the desired resolution. For example, an analog input can be converted with 9 bit resolution, but a TTL input needs only a 1 bit conversion to resolve a high or low digital level.

The input channels consist of four analog sampling switches which use a common sampling capacitor (external capacitor for the 2920, internal capacitor for the 2921). The external capacitor should be 500 pF to yield an offset of less than $-\frac{1}{2}$ LSB. The acquisition time should be approximately six times the RC time constant of the sample and hold circuitry (i.e., 500 pF x 1.5 kΩ = 750 nsecs).

The DAR is used as the successive approximation register, and as such should be cleared at the start of an analog input sequence. An input is selected and sampled using the IN(k) instruction, where k is the selected input (0 to 3). A series of consecutive IN's may be required to charge the sample capacitor to sufficient accuracy. For full 9 bit resolution the acquisition time (TIA) shown in the specifications table should be met. So, the number of consecutive IN's required is a function of the instruction cycle time and therefore a function of the chosen crystal or clock frequency. To calculate the number of IN(k) instructions needed to achieve less than -54 dB of input crosstalk, the following formula can be used:

$$\#\ IN's = (TIA_{max} \times f_{osc})/4$$

where TIA is the input acquisition time and f_{osc} is the crystal or clock frequency.

Once the input has been acquired (sampled and held), the successive approximation conversion sequence can begin. Normally the sign bit is converted first, and then each succeeding bit in order until the 9th bit (LSB) is converted. The digital word in the DAR (cleared at the start of the sequence) is converted by the DAC to a voltage which is one of the comparator inputs. The comparator compares the DAC voltage to the sampled and hold analog level. The comparator's output is routed via the DAR multiplexer to the appropriate bit position of the DAR. This new DAR value will propagate through the DAC to establish the new level to be compared for the bit conversion in the sequence. In between each successive bit conversion a 1.1 μ-sec (2920) or 800 nsec (2921) delay is required to allow the DAC time to settle. This delay is implemented in software by putting no operation (NOP) codes between conversions. Table 6A and B present analog I/O instruction requirements for the 2920 and 2921, respectively. An example input analog instruction conversion sequence, for the 2920, is presented in Table 7.

ANALOG OUTPUTS

An analog output is generated by converting the contents of the DAR into a voltage level using the DAC. The DAC output voltage is routed by the output multiplexer to one of eight internal sample and hold capacitors associated with the eight output buffers. This conversion and outputting is accomplished by a short sequence of instructions. First, the value to be output is moved to the DAR. Next, a short delay (a few NOP's) allows the DAC output voltage to settle. Then, the OUT(k) instruction selects the desired output (k = 0 through 7) and allows the output sample and hold to acquire the DAC output voltage. The sample and hold circuit has a required acquisition time (TOA) that is a function of voltage swing, which is met by a series of OUT (k) instructions. A rule that must be followed in outputting samples is that no outputting should be done while writing to the DAR. An example analog instruction conversion sequence for outputting (based on the analog output requirements presented in table 6-A) is presented in Table 8.

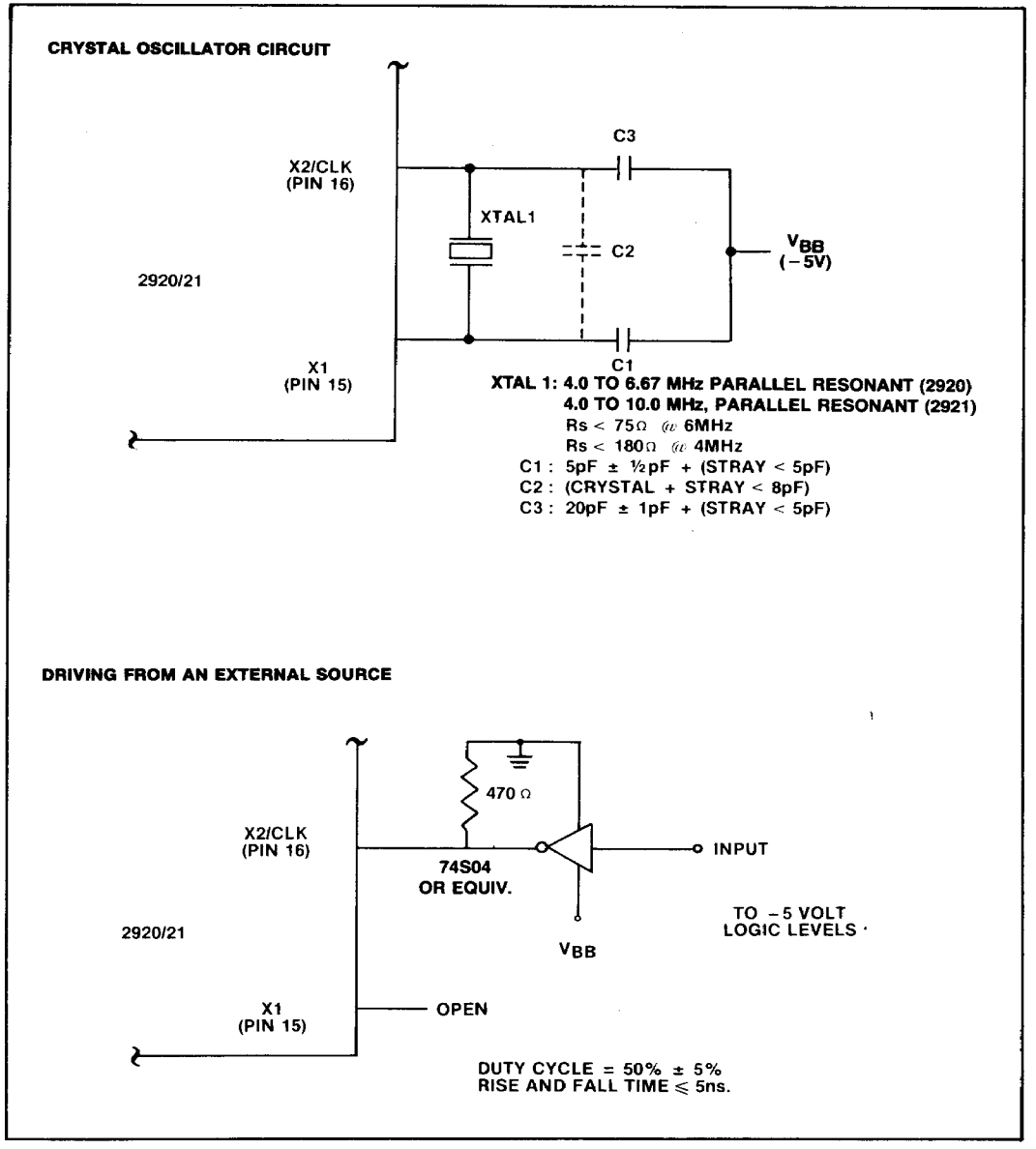


Figure 4. Clocking the 2920/21

Table 6. Signal Processor I/O Requirements

A. 2920 (EPROM)

	ANALOG I/O INSTRUCTION	TIME REQUIRED	INSTRUCTIONS REQUIRED Clock Rate 5 MHz	6.677 MHz Max
INPUT	IN(K)	4 μ secs	5	7
	NOPS BETWEEN CVTS	1100 nsecs	2	2
	NOPS AFTER IN(K) SEQ.	1100 nsecs	2	2
OUTPUT	NOPS AFTER LDA DAR	1100 nsecs	2	2
	OUT(K)	3800 nsecs	5	7

B. 2921 (ROM)

	ANALOG I/O INSTRUCTION	TIME REQUIRED	INSTRUCTIONS REQUIRED Clock Rate 5 MHz	10.0 MHz Max
INPUT	IN(K)	715 nsecs	1	2
	NOPS BETWEEN CVTS	800 nsecs	1	2
	NOPS AFTER IN(K) SEQ.	800 nsecs	1	2
OUTPUT	NOPS AFTER LDA DAR	4900 nsecs	7*	13*
	OUT(K)	800 nsecs	1	2

*This number based on Vref = 2 V. For a Vref = 1 V divide # of NOPS by 2. For any other Vref, # NOPS can be determined by interpolation.

Table 7. Example 2920 Input Sequence (6.67 MHz)

Digital	Analog	Digital	Analog
SUB DAR, DAR	IN(K)	. . . .	CVT5
. . . .	IN(K)	. . . .	NOP
. . . .	IN(K)	. . . .	NOP
. . . .	IN(K)	. . . .	CVT4
. . . .	IN(K)	. . . .	NOP
. . . .	IN(K)	. . . .	NOP
. . . .	IN(K)	. . . .	CVT3
. . . .	NOP	. . . .	NOP
. . . .	NOP	. . . .	NOP
. . . .	CVTS	. . . .	CVT2
. . . .	NOP	. . . .	NOP
. . . .	NOP	. . . .	NOP
. . . .	CVT7	. . . .	CVT1
. . . .	NOP	. . . .	NOP
. . . .	NOP	. . . .	NOP
. . . .	CVT6	. . . .	CVT0
. . . .	NOP		
. . . .	NOP		

Continue Next Column

where " . " equals available digital instruction

Table 8. Example 2920 Output Sequence (6.67 MHz)

Digital	Analog
LDA,DAR,X,R0,	NOP
. . . . .	NOP
. . . . .	NOP
. . . . .	OUT(K)
. . . . .	OUT(K)
. . . . .	OUT(K)
. . . . .	OUT(K)
. . . . .	OUT(K)
. . . . .	OUT(K)
. . . . .	OUT(K)

where K = desired output, "." available digital instruction

Conditional operations should not immediately precede or follow an OUT instruction. Otherwise a CND(K) may affect the value of the Kth output.

TTL Output

The OUT(K) pins can be selected to be either analog out or TTL compatible as seen in Table 9. The analog mode allows the full 9-bit D/A output to be present. The LIM instruction can be used to yield a "0" or "1" decision for the TTL mode. This output can be presented to the OUT(K) pins and is compatible to a single TTL gate or equivalent. The internal threshold required is 1.5 volts for a high level output. An external pullup resistor to V_{CC} is also required.

Table 9. Output Mode for SIGOUT Pins as Function of M1 and M2

M1	M2	OUT Pins
5V	5V	0-7 Analog
5V	-5V	0-3 Analog, 4-7 TTL
-5V	5V	0-3 TTL, 4-7 Analog
-5V	-5V	0-7 TTL

Reference Voltage

The internal D/A converter requires a single positive reference voltage (VREF) to establish its voltage range. This user supplied reference can range from 1V to 2V. The resulting input and output signal voltage range is ±VREF. If the TTL output is required, VREF >1.5V is necessary. The minimum voltage step (LSB) of the D/A converter is $V_{REF}/256$ volts. Voltage variations on V_{REF} will appear as noise to the D/A converter. It is therefore necessary to provide a noise free voltage source for the reference. The input signal voltage range is (±VREF) -½ LSB.

EPROM PROGRAMMING

The 2920 EPROM in the programming mode is arranged as a 1152-by-4-bit memory. Each instruction (24 bits) is loaded as 6 nibbles (4 bits) as seen in Table 3. Figure 6 shows the timing relationships of the signals required to program and verify the EPROM contents. In the program mode all voltages are referenced to V_{BB} which is set to digital ground, thereby allowing TTL logic to be used for controlling the programming cycle. The D pins are bidirectional with the direction controlled by the PROG/VER pin. A high level at the PROG/VER pin switches to input mode, a low to output mode (see Figure 6). This feature allows the programmed data to be verified before going on to the next address.

The internal nibble counter is incremented during the falling edge of INCR. 1152 INCR transitions will complete the full program cycle. To initialize at address (nibble) 0, RST must be pulsed low, then INCR can be issued. From then on, programming is accomplished according to Figure 6.

The RUN/PROG pin must be tied to V_{BB} and VSP should be pulsed between +5V and $+25 \pm 1$V at 15mA maximum. The D pins have an open drain in the output direction.

2920 DEVELOPMENT SUPPORT TOOLS

Support tools for the 2920 are based on the Intellec® Microcomputer Development Systems. A 2920 Software Support Package (SPS-20) consisting of the Signal Processing Applications Software/Compiler (SPAS-20), the 2920 Software Simulator, and the 2920 Assembler is available to facilitate design and development efforts.

The 2920 Signal Processing Applications Software/Compiler is an interactive tool for designing software to be executed on the 2920 Signal Processor. The compiler accepts English-like statements from the user and generates 2920 assembly language code.

The assembler translates symbolic 2920 assembly language programs into the machine operation codes. The user can load the codes into the 2920 simulator or to the Universal EPROM Programmer for programming the 2920 itself.

The simulator, operating entirely in software, allows the user to test and debug 2920 programs. The user can specify input signals, simulate program execution, set up breakpoints, display input and output, display and alter the contents of the 2920 registers and memory locations, and graph the output waveforms.

The compiler, assembler and simulator enable the designer to develop and test an entire program without programming the device. The 2920 designer works at the Intellec® Microcomputer Development System rather than at a breadboard. The development system facilitates the designing and testing of 2920 applications.

The SDK-2920 contains all of the components required to assemble a complete single board microcomputer system for programming and evaluation of the 2920 Analog Signal Processor. The 8085/8041A microcomputer-based program development section allows you to immediately enter programs in 2920 assembly mnemonics, translate them to 2920 object code, and program the on-board 2920 EPROM. The kit supports basic filing options such as up/down loading to/from an Intellec, audio cassette, and line printer. The SDK-2920 also provides the user with a 2920 run mode section allowing real-time execution of a programmed 2920. This section comes complete with BNC connectors and Intel's 2912 PCM line filters required for one input and one output network. The kit supports optional input and output circuitry on the run mode section.

ABSOLUTE MAXIMUM RATINGS

Temperature Under Bias ..... −40°C to +85°C
Storage Temperature ..... −65°C to +150°C
V_{CC} Supply with Respect to GRDD ..... +5.5V
V_{BB} Supply with Respect to GRDD ..... −5.5V
GRDA Supply with Respect to GNDD ..... +/− 0.05V
Analog Inputs with Respect to GRDA ..... −5.5V to +5.5V
Power Dissipation 2920 (at 0°C) 1W
2921 (at 0°C) 1.5W

**NOTICE: Stresses above those listed under "Absolute Maximum Ratings" may cause permanent damage to the device. This is a stress rating only and functional operation of the device at these or any other conditions above those indicated in the operational sections of this specification is not implied. Exposure to absolute maximum rating conditions for extended periods may affect device reliability.*

D.C. CHARACTERISTICS (Run Mode) (T_A = 0°C to +70°C; V_{CC} = 5V ±5%; V_{BB} = −5V ±5%)

ANALOG (IN(K), OUT(K))

Symbol	Parameter	2920 Min	2920 Typ	2920 Max	2921 Min	2921 Typ	2921 Max	Units	Test Conditions
Z_I	Input Impedance	100			200			MΩ	Unsampled input
Z_{SH}	S&H Impedance	0.7	1.2	2.0	5	10	20	MΩ	Series resistance with S&H
	A/D Resolution			9			9	bits	
	Input Differential Linearity		±.75	±2.0		±.5		LSB	2V Vref
	Input Integral Linearity		±.75	±2.0		±1.0	±2.5	LSB	2V Vref
T_A	Aperature		±7.5			±7.5		nsec	
X_I	Input Crosstalk			−54		−60		dB	Input to input
V_{ID}	Input Droop Rate			50			50	μV/μs	500 pF cap.
V_{IR}	Input Voltage Range			± VREF			±VREF	V	Note 1
V_{IOS}	Input Zero Offset	.1				±10		mV	
I_{IZC}	Input Zero Crossing Error		.5	2		0		LSB	2V Vref
G_I	Input Gain	.95	1	1.05	.95	1.0	1.05	V/V	
	Input Gain Error		2			0		%	Ratio of neg. & pos. gains
T_{IA}	Input Acquisition Time[2]	3.6			.7			μs	9 bit resolution
Z_O	Output Impedance		800			800		Ω	During output
I_O	Output Drive Current		350			350		μA	
	D/A Resolution			9			9	bits	
	Output Differential Linearity		±.2	±1		±.5		LSB	2V Vref
	Output Integral Linearity		±.5	±2		±1		LSB	2V Vref
X_O	Output Crosstalk		−59	−45		−59	−45	dB	Output to output
V_{OD}	Output Droop Rate			7			7	μV/μsec	
V_{OR}	Output Voltage Range	−2		+2	−2		+2	V	
V_{OOS}	Output Zero Offset	0	−16	−70		±10	±40	mV	
V_{OZC}	Output Zero Crossing Error			±1			±1	LSB	2V Vref
G_O	Output Gain	.8	.885	1.0	.95	1.0	1.05	V/V	
	Output Gain Error		.55	1.0		—		%	Ratio of neg. & pos. gains
T_{OA}	Output Acquisition Time[3]	4.2			.375		.75	μs	9-bit resolution
G_T	Throughput Gain		.885			1.0		V/V	A/D to D/A
VREF	Voltage Reference Level			2.0			2.0	V	
I_{VR}	Voltage Reference Current		150	220		400		μA	2V Vref
C_L	Max. Capacitance Load			50			50	pF	

Notes: 1. V_{IR} = ± Vref for accurate conversion. Input levels outside of ±Vref will be converted as ± full scale.
2. For 500 pF capacitor (2920 only)
3. Minimum corresponds to 1V Vref, maximum to 2V Vref.

DIGITAL ($\overline{OF}$,$\overline{CCLK}$,$\overline{RST}$/$\overline{EOP}$,X2/CLK, OUT(K) in TTL Mode)

Symbol	Parameter	2920 Min	2920 Typ	2920 Max	2921 Min	2921 Typ	2921 Max	Units	Test Conditions
I_{IL}	Low Level Input Current			10			10	μA	VIN < VIL Driving $\overline{RST}$
I_{IH}	High Level Input Current			10			10	μA	VIN < VIH Driving $\overline{RST}$
V_{IL}	Input Low Voltage			0.8			0.8	V	DC trigger level
V_{IH}	Input High Voltage	2.0			2.0			V	DC trigger level
V_{IXL}	Input Low Voltage, X2/CLK	−5.0		−4.5	−5.4	−5	−4.6	V	
V_{IXH}	Input High Voltage, X2/CLK	−.3		0	−2.0	0	0	V	
I_{OL}	Output High Current	1.7	2		1.7			mA	V_{OL} = .4V
I_{OH}	Output Low Current			10			10	μA	

POWER DISSIPATION

I_{CC}	Operating Current		30	50		50	80	mA	V_{CC} = 5.25V
I_{BB}	Operating Current		110	150		150	200	mA	V_{BB} = −5.25V

A.C. CHARACTERISTICS (Run Mode)

F_{OSC}		4.0		6.67	1		10	MHz	Oscillator frequency
T_{CYC}	Instruction Cycle Period	600		1000	400		4000	ns	4 clock cycles
$T_{\overline{CE}}$	Cycle Start to EOP Valid			255	100	200	240	ns	
T_{EW}	EOP Pulse Width	TCYC -150			360		450	ns	
T_{OW}	OF Pulse Width	TCYC -100			360		450	ns	
T_{CW}	CCLK Pulse Width	TCYC -65			360		450	ns	
T_{CO}	Cycle Start to OF Valid			TCYC +95	360		450	ns	

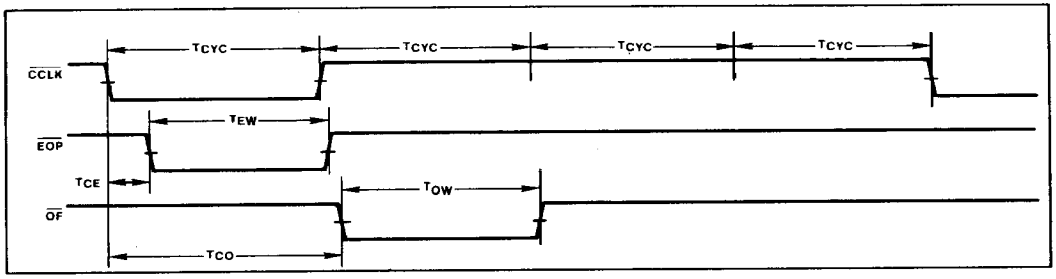


Figure 5. Run Mode Timing

D.C. CHARACTERISTICS (Program/Verify Mode) (T_A = 0°C to +70°C; V_{SS} =5V; V_{BB} = 0V)

Symbol	Parameter	2920 Min	2920 Typ	2920 Max	2921 Min	2921 Typ	2921 Max	Units	Test Conditions
I_{TL}	Low Level Input Current			10			10	μA	$V_{IN} < V_{IL}$
I_{IH}	High Level Input Current			10			10	μA	$V_{IN} < V_{IH}$
V_{IL}	Input Low Voltage			0.8			0.8	V	
V_{IH}	Input High Voltage	2.0			2.0			V	
V_{OL}	Output Low Voltage			0.8			0.8	V	I_{OL} = 2.5mA
I_{SP}	Program Pulse Current			16	—	—	—	mA	Data Input=0000 (2920 Only)
V_{P1}	Program Pulse ON Voltage	24	25	26	—	—	—	V	(2920 Only)
V_{P2}	Program Pulse OFF Voltage		5		—	—	—	V	(2920 Only)
POWER DISSIPATION (Program/Verify Mode)									
I_{SS}	Operating Current		100			—		mA	V_{SS} = 5V+10%

A.C. CHARACTERISTICS (Program/Verify Mode) (T_A = 0°C to +70°C; V_{SS} =5V; V_{BB} = 0V)

T_{RW}	Reset Pulse Width	1000			1000			ns	
T_{RS}	Reset to Increment Set-up	200			200			ns	
T_{RH}	Reset Hold	300			300			ns	
T_{IW}	Increment Pulse Width	1			1			μs	
T_{VP}	Data in Set-Up to Prog. Pulse	2			2			μs	
T_{PW}	Program Pulse Width (2920)	50		55	—	—	—	ms	
	Verify Hold Time (2921)	—	—	—	50		55	ms	
T_{PV}[1]	Program to Verify Settling	1			1			μs	
T_{ACC}	Verify Access Time	20			20			μs	
T_{VI}	End of Verify to Increment	100			100			ns	

Note: 1. Vpp must not undershoot 5V by more than 0.5V. Add undershoot settling time to T_{PV}.
***25V program pulse must NOT be applied to 2921.**

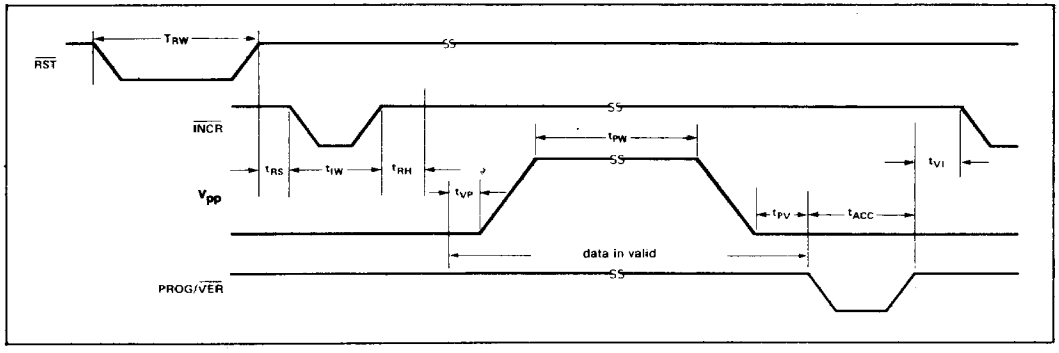


Figure 6. 2920 Program/Verify, 2921 Verify Timing